LaTeX Mini-Tutorial

This is a very brief introduction to using the free LaTeX typesetting system. It is assumed that you have installed a typical LaTeX system such as MiKTeX for Windows (available at http://miktex.org), though the general ideas apply to any LaTeX system for any operating system (e.g. TeX Live for Mac OS X or Linux, available at http://tug.org/texlive/). More LaTeX documentation can be found at http://www.tug.org/begin.html

Introduction

In LaTeX, plain text files (called the *source files*) contain the commands that are used for producing a document in a printable format, such as PDF or PostScript. In MiKTeX, these text files are typed into the TeXworks text editor, then compiled into a PDF file by clicking the green arrow button in the main menubar. The PDF file is displayed in a separate window. The LaTeX files are typically given a .tex extension, for example myfile.tex. In this case the resulting PDF file would be named myfile.pdf. A standard LaTeX source file has the following structure:

```
\documentclass{...}

<some commands to load certain packages and set document settings>

\begin{document}

<main text of the document>

\end{document}
```

The first line of a LaTeX source file is the `\documentclass` command, which is used as follows:

```
\documentclass[options]{name of class}
```

The [`options`] part is optional, though usually you end up using it. The {`name of class`} part is mandatory; you must supply the name of the type of *class* of document you want. Some standard document classes are:

Class	Description
article	Short documents, e.g. reports, essays, papers.
exam	Exams and quizzes.
letter	Letters, either personal or business.
book	Books, with chapters, sections, subsections, table of contents, etc.

For example, if you wanted to write a typical short document, using an 11 point font and the standard 8.5"×11" US letter paper size, then the first line in your LaTeX source file would be:

```
\documentclass[11pt,letterpaper]{article}
```

The section of your LaTeX source file between the `\documentclass` command and the `\begin{document}` command is called the *preamble*. This is where you can tell LaTeXto load extra *packages* for more functionality beyond the default. For example, the default page margins are fairly large, so you can load the `geometry` package to make the horizontal margins (`hmargin`) and the vertical margins (`vmargin`) smaller, say, 1 inch each:

```
\usepackage[hmargin=1in,vmargin=1in]{geometry}
```

In general, packages are loaded with the following syntax:

```
\usepackage[options]{name of package}
```

If you exclude the optional [`options`] part, then LaTeX will use the defaults for the package you load.

Simple Example

After loading any necessary packages in the preamble, you can start typing the text for the body of your document. To do so, put your text between the `\begin{document}` and `\end{document}` commands. Here is an example of a complete LaTeX source file (save as hello.tex) which creates just one paragraph of text:

```
\documentclass[letterpaper,11pt]{article}
\usepackage[hmargin=1in,vmargin=1in]{geometry}
\begin{document}
Hello! This is my first \LaTeX{} document. I will learn more about this typesetting
system and write lots of math documents with it. Wish me luck!
\end{document}
```

The PDF document (hello.pdf) created by clicking the green arrow button in MiKTeX (or by compiling it with the command `pdflatex hello.tex` in a DOS command window or Linux terminal) will look like this:

Hello! This is my first LaTeX document. I will learn more about this typesetting system and write lots of math documents with it. Wish me luck!

1
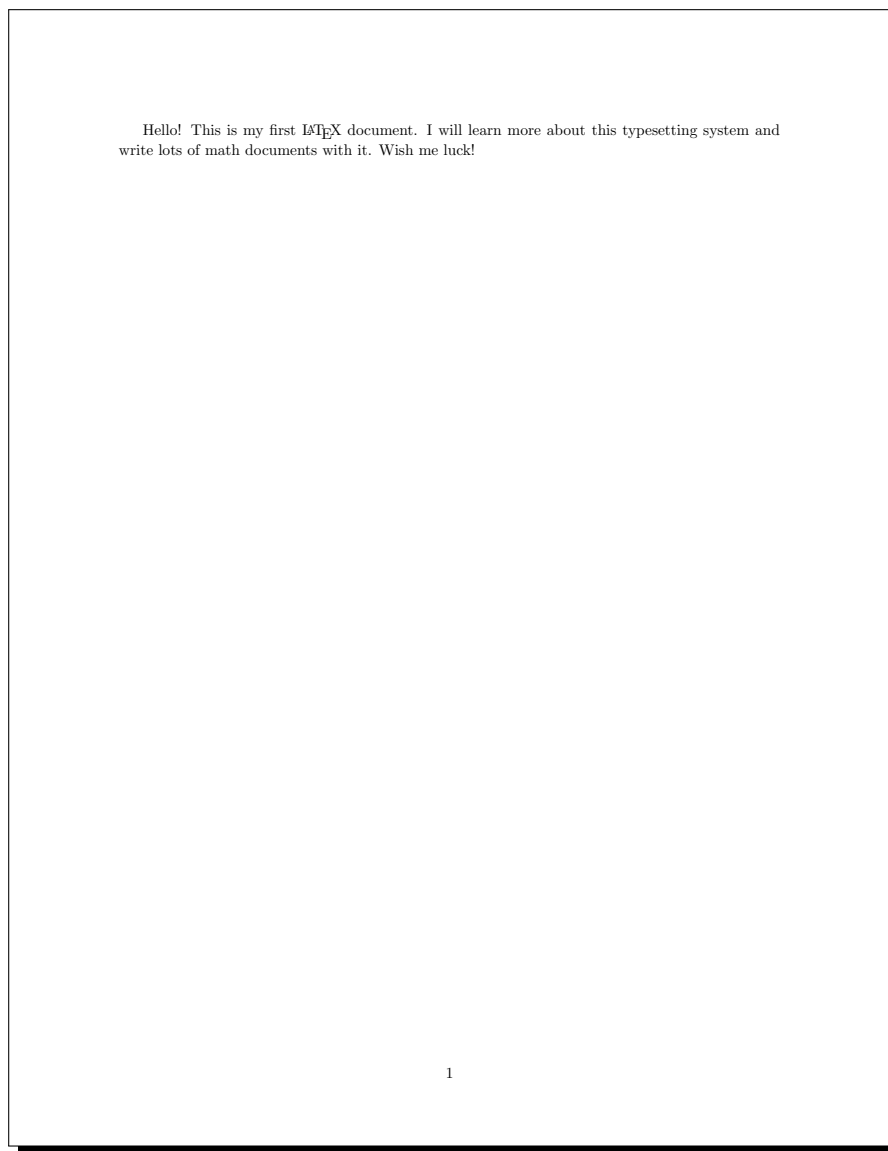
Notice that the first line of the paragraph is automatically indented. To suppress this indentation, use the `\noindent` command (followed by a space) at the beginning of the paragraph. Notice also that in the source file the first line of the paragraph ends at the word "typesetting," but not in the PDF output file. This is because LaTeX

treats a single newline (carriage return) and multiple spaces as a single space, and it will automatically wrap lines for you. To force text on a new line, use the \\ command at the point where you want the current line to end:

```
This will be on line 1.\\This will be on line 2.
```

This produces the output:

> This will be on line 1.
> This will be on line 2.

Use blank lines (or the \par command) to start new paragraphs:

```
Hello! This is my first \LaTeX{} document. I will learn more about this typesetting
system and write lots of math documents with it. Wish me luck!

I now know how to start new paragraphs. This second paragraph is going to be fun!
```

This produces the output:

> Hello! This is my first LaTeX document. I will learn more about this typesetting system and write lots of math documents with it. Wish me luck!
>
> I now know how to start new paragraphs. This second paragraph is going to be fun!

Text Formatting

The following commands show how to change the appearance of text:

Appearance	Command	Example	Output
Bold	\textbf{...}	\textbf{bold text}	**bold text**
Italic	\emph{...}	\emph{italic text}	*italic text*
Bold italic	\textbf{\emph{...}}	\textbf{\emph{bold italic text}}	***bold italic text***
Underline	\underline{...}	\underline{underline text}	underline text
Small caps	\textsc{...}	\textsc{small caps text}	SMALL CAPS TEXT
Monospaced	\texttt{...}	\texttt{monospaced text}	monospaced text
Sans serif	\textsf{...}	\textsf{sans serif text}	sans serif text
Superscript	^{...}	a^{superscript}	a^{superscript}
Tiny	\tiny{...}	\tiny{tiny text}	tiny text
Superscript size	\scriptsize{...}	\scriptsize{superscript size text}	superscript size text
Footnote size	\footnotesize{...}	\footnotesize{footnote size text}	footnote size text
Small	\small{...}	\small{small text}	small text
Normal size	\normalsize{...}	\normalsize{normal size text}	normal size text
Large	\large{...}	\large{large text}	large text
Larger	\Large{...}	\Large{larger text}	larger text
Largest	\LARGE{...}	\LARGE{largest text}	largest text
Huge	\huge{...}	\huge{huge text}	huge text
Hugest	\Huge{...}	\Huge{hugest text}	hugest text
Blue[1]	\textcolor{blue}{...}	\textcolor{blue}{blue text}	blue text

---

[1] This requires loading the xcolor package in the preamble: \usepackage{xcolor}
You can of course use other colors, e.g. red, yellow, green, magenta, brown, etc. You can also define your own colors based on different color models, e.g. RGB, CMYK, HTML. For more information see the xcolor package documentation at http://mirror.ctan.org/macros/latex/contrib/xcolor/xcolor.pdf

Special Characters

There are some characters which have special meaning in LaTeX, e.g. \. The table below shows how to use these characters in normal text mode. To use the `\symbol{...}` commands for the last five characters in the table, you need to load the `fontenc` package in the preamble with the T1 character encoding: `\usepackage[T1]{fontenc}`

Character	Command	Example	Output
\	`\textbackslash`	`This is a backslash:  \textbackslash`	This is a backslash: \
%	`\%`	`This is 50\%.`	This is 50%.
$	`\$`	`This is \$50.`	This is $50.
#	`\#`	`This is \#50.`	This is #50.
&	`\&`	`Abbott \& Costello`	Abbott & Costello
^	`\symbol{94}`	`This is a caret:  \symbol{94}`	This is a caret: ^
~	`\symbol{126}`	`This is a tilde:  \symbol{126}`	This is a tilde: ~
{	`\symbol{123}`	`This is a left brace:  \symbol{123}`	This is a left brace: {
}	`\symbol{125}`	`This is a right brace:  \symbol{125}`	This is a right brace: }
_	`\symbol{95}`	`This is an underscore:  \symbol{95}`	This is an underscore: _

Lists

Use the `enumerate` environment to create a numbered list. This starts with the `\begin{enumerate}` command, followed by an `\item` command for each numbered item in the list, and ends with the `\end{enumerate}` command. The example below on the left shows code for creating a list of 3 items, with the output shown on the right:

```
\begin{enumerate}
 \item This is item 1.
 \item This is the second item.
 \item This is item 3.
\end{enumerate}
```

1. This is item 1.

2. This is the second item.

3. This is item 3.

Notice that you do not have to manually number the list items; LaTeX does the numbering for you. You can also create a sub-list within a list item by putting another `enumerate` environment inside that item. For example, the code below on the left adds two parts, (a) and (b), to item 2 in our above list; the output is shown on the right:

```
\begin{enumerate}
 \item This is item 1.
 \item This is the second item.
  \begin{enumerate}
   \item This part will be easy.
   \item This part will be hard!
  \end{enumerate}
 \item This is item 3.
\end{enumerate}
```

1. This is item 1.

2. This is the second item.

   (a) This part will be easy.

   (b) This part will be hard!

3. This is item 3.

Notice that LaTeX automatically knew to label the sub-items in item 2 with letters instead of numbers. You can continue this nesting of `enumerate` environments within `enumerate` environments to produce sub-sub-lists, and so on. If you were to start a new `enumerate` environment completely *outside* any previous `enumerate` environment, then the list created by that new `enumerate` environment would have its numbering start back at 1. So, for example, if you were creating an exam or quiz, you would typically have one main `enumerate` environment in your document (with items numbered 1, 2, 3, etc), and any others would be inside that main `enumerate` environment. You would use completely separate `enumerate` environments if, for example, you made an exam with multiple sections.

To create bullet (unnumbered) lists, use the `itemize` environment:

```
\begin{itemize}
 \item This is the first bullet item.
 \item This is the second bullet item.
 \item This is bullet item 3.
\end{itemize}
```

- This is the first bullet item.
- This is the second bullet item.
- This is bullet item 3.

As with numbered lists, you can create sub-lists of bullet lists, in this case by nesting an `itemize` environment within an `itemize` environment.

Tables

Use the `tabular` environment to create tables. In its most basic form, the format is shown below:

```
\begin{tabular}{column specifications}
 row 1\\
 row 2\\
 ...
 row n-1\\
 row n
\end{tabular}
```

The column specifications consist of a sequence of characters from among `c`, `l`, and `r`, one character for each column in the table, which indicates how the text in that column is justified: `c` for centered, `l` for left-justified, and `r` for right-justified.

The format for the rows is to separate the column entries by an ampersand (&) and terminate the row with a double backslash (\\). For example, a basic table with 3 columns and 3 rows is shown below, with the first column centered, the second column left-justified, and the third column right-justified:

```
\begin{tabular}{clr}
 Column 1 & Column 2 & Column 3\\
 This & is & a row\\
 And this & will be & another row
\end{tabular}
```

Column 1	Column 2	Column 3
This	is	a row
And this	will be	another row

By default there are no borders around the table or any lines to separate columns and rows. Vertical lines for the columns are created by putting a vertical bar | in the desired positions in the column specifications, while horizontal lines are created with the `\hline` command in the desired positions among the rows. The `\hline` command does not need a terminating double backslash. For example, here is the above table with borders around the outside of the table, a vertical line between columns 1 and 2, and horizontal lines between the rows:

```
\begin{tabular}{|c|lr|}
\hline
 Column 1 & Column 2 & Column 3\\
\hline
 This & is & a row\\
\hline
 And this & will be & another row\\
\hline
\end{tabular}
```

Column 1	Column 2	Column 3
This	is	a row
And this	will be	another row

Spacing

Horizontal spaces of various sizes can be created with the following commands:

Command	Example	Output
\,	I\,I	II
\:	I\:I	II
\;	I\;I	II
~	I~I	II

Command	Example	Output
\enskip	I\enskip I	I I
\quad	I\quad I	I I
\qquad	I\qquad I	I I
\hspace{*length*}	I\hspace{0.5in}I	I I

The \hspace{*length*} command can be given lengths in different units, e.g. \hspace{1in}, \hspace{1.3cm}, \hspace{5mm}, \hspace{12pt} (72 pt equals 1 inch). It can also be given negative lengths to move backwards, e.g. \hspace{-0.75in}. The maximum horizontal length on a page is called \textwidth.

Vertical spacing can be created with the \vspace{*length*} command, which can be placed at the end of paragraphs and various environments (e.g. tables, lists). For example, in the list we created earlier, here is how to put a 0.5 inch vertical space between items 1 and 2:

```
\begin{enumerate}
 \item This is item 1.\vspace{0.5in}
 \item This is the second item.
 \item This is item 3.
\end{enumerate}
```

1. This is item 1.

2. This is the second item.

3. This is item 3.

To add vertical space below an entire environment, put your \vspace command after the \end{...} command of the environment. For example, to add a 1 inch vertical space after an enumerate environment, you would use \end{enumerate}\vspace{1in}. You can move upwards by using a negative length in the \vspace command. This is helpful when you want to remove unwanted vertical space that was created automatically by some environment.

To add vertical space *inside* tables or other environments where rows are terminated by a double backslash, you can use the [*length*] command after the double backslash. For example, here is how you could add a 5 millimeter vertical space between the second and third rows in our table example from before:

```
\begin{tabular}{clr}
 Column 1 & Column 2 & Column 3\\
 This & is & a row\\[5mm]
 And this & will be & another row
\end{tabular}
```

Column 1	Column 2	Column 3
This	is	a row
And this	will be	another row

Boxes

Use the \fbox{*text*} command to put a framed box around a small amount of text (up to one line). To box a paragraph of text, use the \parbox{*width*}{*paragraph*} command to put an invisible box *width* units wide around the paragraph, then put that inside an \fbox. You can use the special length \textwidth for the width:

```
I'm going to box \fbox{this text} first.

\fbox{\parbox{\textwidth}{This whole
paragraph will be boxed. This will make it
seem as if it is very important.}}
```

I'm going to box $\boxed{\text{this text}}$ first.

This whole paragraph will be boxed. This will make it seem as if it is very important.

Positioning

To control the horizontal positioning of a fragment of text or of an environment (e.g. a table), you can enclose the object within the following commands: `\begin{flushleft}` and `\end{flushleft}` for alignment on the left margin; `\begin{flushright}` and `\end{flushright}` for alignment on the right margin; and `\begin{center}` and `\end{center}` to center the object. For example:

```
\begin{center}
 \Large{\textbf{Here is a centered title}}
\end{center}
This line is not centered.
```

> ## Here is a centered title
> This line is not centered.

Pagination

LaTeX will automatically create new pages when needed. You can force a new page with the `\newpage` command on a line by itself. You can disable page numbering with the `\pagestyle{empty}` command in the preamble.

Miscellaneous

Some extra symbols in normal text mode:

Symbol	Command	Example	Output
fancy double quotes	`''text''`	`A ''fancy quote''`	A "fancy quote"
accent acute	`\'{character}`	`caf\'{e}`	café
accent grave	`\`{character}`	`tr\`{e}s gauche`	très gauche
accent circumflex	`\^{character}`	`L'H\^{o}pital's Rule`	L'Hôpital's Rule
umlaut	`\"{character}`	`M\"{o}bius strip`	Möbius strip
cedilla	`\c{character}`	`Fran\c{c}ois Truffaut`	François Truffaut
diacritical tilde	`\~{character}`	`pi\~{n}ata`	piñata
cents[2]	`\textcent`	`50\textcent`	50¢

Examples of making horizontal lines of various lengths:

```
One inch line, 0.5 pts thick: \rule{1in}{0.5pt}

Same as above, 3 pts lower:
\rule[-3pt]{1in}{0.5pt}

Line to right margin: \hrulefill

Put a line across the page:\\
\hrule
```

> One inch line, 0.5 pts thick: _____
> Same as above, 3 pts lower: _____
> Line to right margin: _____
> Put a line across the page:
> _____

You can import external graphics by putting the command `\usepackage{graphicx}` in the preamble and putting the `\includegraphics[scale=scale factor]{file name}` command in the document body where you want the image to appear (*scale factor* is a ratio greater than 0). For example, suppose the image file oski.jpg is in the same directory as your LaTeX source file, and you want it to appear along the right margin at 85% its usual size:

```
\begin{flushright}
 \includegraphics[scale=0.85]{oski.jpg}
\end{flushright}
```

---

[2]Requires the `textcomp` package to be loaded in the preamble: `\use{textcomp}`

Mathematics

So far all the commands discussed are for LaTeX' normal text mode. Mathematical symbols and equations require you to be in *math mode*. There are two ways to enter math mode:

1. *Inline* math: This is used when the mathematics is to appear in a paragraph with normal text. To use this, enclose the mathematics between two dollar sign symbols (`$...$`) in a normal text paragraph.

2. *Display* math: This is used when the mathematics is to appear in a separate environment, not part of a normal text paragraph. The most basic math environment is the `displaymath` environment. By default, the various math environments are centered horizontally, apart from normal text paragraphs.

Here is an example of mathematics in both inline and display modes:

```
This is $x^2 =\frac{1}{4}$ in inline mode.
Here it is in display mode:
\begin{displaymath}
 x^2 = \frac{1}{4}
\end{displaymath}
```

This is $x^2 = \frac{1}{4}$ in inline mode. Here it is in display mode:
$$x^2 = \frac{1}{4}$$

The `displaymath` environment is built in to LaTeX and requires no extra packages to be loaded. However, it is likely that you will want to use some of the other math environments provided by the `amsmath` package, which we will assume from now on you have loaded in the preamble:[3] `\usepackage{amsmath}`

Here are some common math mode commands and symbols (enclose between dollar signs for inline mode):

Math	Example	Output
Addition	`a + b`	$a + b$
Subtraction	`a - b`	$a - b$
Multiplication	`a \times b`	$a \times b$
Division	`a \div b`	$a \div b$
Equality	`a = b`	$a = b$
Not equal	`a \ne b`	$a \ne b$
Greater than	`a > b`	$a > b$
Less than	`a < b`	$a < b$
Fraction	`\frac{a}{b}`	$\frac{a}{b}$
Exponent	`a^b`	a^b
Subscript	`a_b`	a_b
Square root	`\sqrt{a}`	$\sqrt{a}$
Plus or minus	`\pm`	$\pm$
Infinity	`\infty`	∞
Degrees	`45^{\circ}`	$45°$
Angle	`\angle A`	$\angle A$
Triangle	`\triangle ABC`	$\triangle ABC$
Parallel	`l \parallel m`	$l \parallel m$
Perpendicular	`l \perp m`	$l \perp m$
Intersection	`A \cap B`	$A \cap B$
Union	`A \cup B`	$A \cup B$
Subset	`A \subset B`	$A \subset B$
Empty set	`\emptyset`	$\emptyset$
Equivalent	`y \equiv x`	$y \equiv x$
Approximately	`y \approx x`	$y \approx x$
Similar	`y \sim x`	$y \sim x$

Math	Example	Output
Greater than or equal to	`a \ge b`	$a \ge b$
Less than or equal to	`a \le b`	$a \le b$
Implies	`P \Rightarrow Q`	$P \Rightarrow Q$
Two-way implication	`P \Leftrightarrow Q`	$P \Leftrightarrow Q$
Summation	`\sum_{n=1}^{N} a_n`	$\sum_{n=1}^{N} a_n$
Limit	`\lim_{x \to a} f(x)`	$\lim_{x \to a} f(x)$
Derivative	`f'(x)`	$f'(x)$
Second derivative	`f''(x)`	$f''(x)$
Partial derivative	`\partial f`	∂f
Indefinite integral	`\int f(x)~dx`	$\int f(x)\, dx$
Definite integral	`\int_{a}^{b} f(x)~dx`	$\int_a^b f(x)\, dx$
Double integral	`\iint\limits_R f~dA`	$\iint\limits_R f\, dA$
Triple integral	`\iiint\limits_S f~dV`	$\iiint\limits_S f\, dV$
Multiple integral	`\idotsint\limits_V`	$\int \cdots \int_V$
Line integral	`\oint_C f~ds`	$\oint_C f\, ds$

[3]Use `\usepackage[fleqn]{amsmath}` to make the math environments left-aligned instead of centered.

The above list barely scratches the surface of the math symbols available in LaTeX.[4] There are also commands for some common math functions, operators, and Greek letters in math mode:

Command	Output	Command	Output	Command	Output	Command	Output
`\sin x`	$\sin x$	`\log x`	$\log x$	`\alpha`	α	`\mu`	μ
`\cos x`	$\cos x$	`\ln x`	$\ln x$	`\beta`	β	`\pi`	π
`\tan x`	$\tan x$	`\log_b x`	$\log_b x$	`\gamma`	γ	`\rho`	ρ
`\csc x`	$\csc x$	`\dot{x}`	$\dot{x}$	`\Gamma`	Γ	`\sigma`	σ
`\sec x`	$\sec x$	`\ddot{x}`	$\ddot{x}$	`\delta`	δ	`\Sigma`	Σ
`\cot x`	$\cot x$	`\bar{x}`	$\bar{x}$	`\Delta`	Δ	`\tau`	τ
`\arcsin x`	$\arcsin x$	`\tilde{x}`	$\tilde{x}$	`\epsilon`	ϵ	`\phi`	ϕ
`\arccos x`	$\arccos x$	`\hat{x}`	$\hat{x}$	`\zeta`	ζ	`\Phi`	Φ
`\arctan x`	$\arctan x$	`vec{v} \cdot vec{w}`	$\vec{v} \cdot \vec{w}$	`\eta`	η	`\chi`	χ
`\sinh x`	$\sinh x$	`\overline{AB}`	$\overline{AB}$	`\theta`	θ	`\psi`	ψ
`\cosh x`	$\cosh x$	`\overrightarrow{AB}`	$\overrightarrow{AB}$	`\kappa`	κ	`\omega`	ω
`\tanh x`	$\tanh x$	`\nabla`	∇	`\lambda`	λ	`\Omega`	Ω

Normal text in math mode is italicized and spaces are ignored. You can use ~ for spacing in math mode, or (preferably) use the `\text{`*`some text`*`}` command. Compare how text appears in text mode and in math mode:

```
Bad: sin x is a function\\
Terrible: $sin x is a function$\\
Good: $\sin x ~\text{is a function}$\\
Good: $\sin x$ is a function
```

Bad: sin x is a function
Terrible: $sinxisafunction$
Good: $\sin x$ is a function
Good: $\sin x$ is a function

You may have noticed in the first math example on the previous page that fractions appear larger in display math mode than in inline math mode. The same is true for math symbols such as `\sum`, `\lim`, and `\int`. You can force display math mode sizes for those symbols in inline math mode by using `\dfrac` instead of `\frac`, and by preceding the other symbols by `\displaystyle`, as the following inline math example shows:

```
Fraction: $\frac{dy}{dx} = \dfrac{dy}{dx}$\\
Sum: $\sum_{n=1}^{\infty} = \displaystyle\sum_{n=1}^{\infty}$\\
Limit: $\lim_{n \to \infty} = \displaystyle\lim_{n \to \infty}$\\
Integral: $\int_a^b = \displaystyle\int_a^b$
```

Fraction: $\frac{dy}{dx} = \dfrac{dy}{dx}$
Sum: $\sum_{n=1}^{\infty} = \displaystyle\sum_{n=1}^{\infty}$
Limit: $\lim_{n\to\infty} = \displaystyle\lim_{n\to\infty}$
Integral: $\int_a^b = \displaystyle\int_a^b$

Here is an example of creating matrices and determinants:

```
\begin{displaymath}
 A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix},~
 B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix},~
 |A| = \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix}
\end{displaymath}
```

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \ B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}, \ |A| = \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix}$$

Delimiters such as parentheses, braces, brackets and vertical bars are not automatically sized to fit their contents (e.g. the parentheses in $(\frac{1}{2})$ do not fit the enclosed fraction). To fix this, use the `\left` and `\right` commands:

---

[4]See *The Comprehensive LaTeX Symbol List*: http://ctan.org/tex-archive/info/symbols/comprehensive/symbols-letter.pdf

```
\begin{displaymath}
 \left( \frac{1}{2} \right) ,~
 \left\lbrace e^{x^2 + y^2} \right\rbrace ,~
 \left\lbrack \iint f~dS \right\rbrack ,~
 \left| \frac{\partial f}{\partial x} \right|
\end{displaymath}
```

$$\left(\frac{1}{2} \right), \; \left\{ e^{x^2+y^2} \right\}, \; \left[\iint f \, dS \right], \; \left| \frac{\partial f}{\partial x} \right|$$

The displaymath environment can display only a single line. The amsmath package provides several multiline math environments, such as the align* environment,[5] which aligns multiple lines (each terminated by a double backslash) at an *anchor*, which is preceded by an ampersand. The align environment does the same and labels each line with a number. The example below uses align* and align, each with lines aligned at an equals sign:

```
\begin{align*}
 f(x) &= e^{x-1}\\
 \int_1^2 f(x)\;dx &= e^{x-1} ~\Bigr|_1^2\\
 &= e - 1
\end{align*}
%This is a comment and will be ignored
\begin{align}
 2x - 4y - 7z + 8w &= \pi\\
 3x + 5y + 9z &= 213
\end{align}
```

$$f(x) = e^{x-1}$$
$$\int_1^2 f(x) \, dx = e^{x-1} \Big|_1^2$$
$$= e - 1$$

$$2x - 4y - 7z + 8w = \pi \tag{1}$$
$$3x + 5y + 9z = 213 \tag{2}$$

Finally, here is a full example of a math quiz (save as mathquiz.tex then compile it into a PDF). Good luck!

```
\documentclass[letterpaper,11pt]{article}
\usepackage[hmargin=1in,vmargin=1in]{geometry}
\usepackage{amsmath}
\pagestyle{empty}
\begin{document}
Math Quiz \hspace{1in} Name: \hrulefill
\begin{enumerate}
 \item Suppose that $ax^2 + bx + c = 0$. Show that
  \begin{displaymath}
   x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} ~.
  \end{displaymath}\vspace{0.75in}
 \item Solve the following system of equations:
  \begin{align*}
   2x - 4y ~&=~ \sum_{n=0}^1 \ln \left( e^3 \right)\\
   3x + 5y ~&=~ \int_0^1 2x\;dx ~+~ \lim_{x \to -\infty} \frac{\sin x}{\sqrt[3]{x}}
  \end{align*}\vspace{2in}
 \item Draw a right triangle $\triangle ABC$ with $C = 90^{\circ}$.\vspace{1in}
 \item Answer the following questions about the function $f(\theta) = \cos \theta$:
  \begin{enumerate}
   \item What is $f'(\theta)$?\vspace{0.75in}
   \item What is $\lim_{\theta \to 0} f(\theta)$?
  \end{enumerate}
\end{enumerate}
\end{document}
```

---

[5] A good discussion of other such environments is at http://mirror.ctan.org/info/math/voss/mathmode/Mathmode.pdf